ADA115186

Semiannual Technical Summary

Multi-Dimensional Signal Processing Research Program

31 March 1982

Prepared for the Department of the Air Force
under Electronic Systems Division Contract F19628-80-C-0002 by

Lincoln Laboratory

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

LEXINGTON, MASSACHUSETTS

The work reported in this document was performed at Lincoln Laboratory, a center for research operated by Massachusetts Institute of Technology, with the support of the [illegible] Air Force Rome Air Development Center under Contract F19628-[illegible]. A part of this work was supported by the Rome Air Development Center.

This report may be reproduced to satisfy needs of U.S. Government agencies.

The views and conclusions contained in this document are those of the contractor and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the United States Government.

The Public Affairs Office has reviewed this report, and it is releasable to the National Technical Information Service, where it will be available to the general public, including foreign nationals.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

Joseph C. Sajek

Joseph C. Sajek
[illegible] Lincoln Laboratory Project Officer

12

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

LINCOLN LABORATORY

MULTI-DIMENSIONAL SIGNAL PROCESSING RESEARCH PROGRAM

SEMIANNUAL TECHNICAL SUMMARY REPORT
TO THE
ROME AIR DEVELOPMENT CENTER

1 OCTOBER 1981 — 31 MARCH 1982

ISSUED 1 JULY 1982

Approved for public release; distribution unlimited.

LEXINGTON MASSACHUSETTS

ABSTRACT

This Semiannual Technical Summary covers the period 1 October 1981 through 31 March 1982. It describes the significant results of the Lincoln Laboratory Multi-Dimensional Signal Processing Research Program sponsored by the Rome Air Development Center, in the areas of image segmentation, classification, target detection, and adaptive contrast enhancement.

TABLE OF CONTENTS

MULTI-DIMENSIONAL SIGNAL PROCESSING RESEARCH PROGRAM

1. INTRODUCTION AND SUMMARY

The Lincoln Laboratory Multi-Dimensional Signal Processing Research Program was initiated in FY 80 as a research effort directed toward the development and understanding of the theory of digital processing of multidimensional signals and its applications to real-time image processing and analysis. A specific long-range application is the automated processing of aerial reconnaissance imagery. Current research projects which support this long-range goal are image modeling for segmentation, classification, and target detection, techniques for adaptive contrast enhancement, and multiprocessor architectures for implementing image-processing algorithms. In this Semiannual Technical Summary we shall describe in detail our work to date on the detection of anomalous areas using an image-modeling approach.

In the area of adaptive contrast enhancement we have recently completed the initial development of a software module for implementing the algorithms previously developed under this program. This software module was written in the programming language "C" to facilitate its transfer to the RADC Automatic Feature Extraction System (AFES), which utilizes a UNIX-based operating system. We are currently in the process of transferring this software module and its documentation to RADC/IRRE for testing.

We have begun the paper development of a strawman multi-processor architecture for image-processing applications. We have been looking at generic image-processing operations such as linear filtering, median filtering, nonlinearities, segmentation, edge detection, and Fourier transformation. Based on operations such as these, it seems that straightforward data partitioning techniques can be used in most cases to fit the operation into a multi-processor implementation. We plan to continue refining our architectural approach during the remainder of FY 82 to incorporate other image operations and to address the problems of programmability, I/O, inter-processor communications, and required performance levels.

The remainder of this report is devoted to a comprehensive treatment of both our theoretical and empirical results in the area of target detection within aerial reconnaissance imagery.

2. TARGET DETECTION BY PARAMETER TRACKING

This component of our research relates to the problem of detecting targets (i.e., anomalous areas) in aerial photographs and is an outgrowth of our work in image modeling, segmentation, and classification. We can loosely define the target-detection problem as the detection of man-made objects in a textured background (e.g., trees, grass, fields, etc.).

Usually, in detection theory the target (or signal) is added to the background (or noise), and filtering procedures are well established for increasing the signal-to-noise ratio. In image processing, however, the target pixels replace the background pixels. Motivated by this observation, we model an aerial photograph as a 2-D random process which is the output of a continuously space-varying linear filter. We assume that a textured background (i.e., the 2-D filter representing the background) is slowly varying, while an anomalous area of the image reflects a quick change in the filter characteristics.

One possible interpretation of the target-detection problem then consists of detecting large changes in 2-D filter parameters. That is, we "track" the parameters of the model; when a target area is reached, the parameters tend to change quickly, indicating the presence of a target. Furthermore, we shall assume no knowledge of the target or background except that the target's spectral characteristics differ from that of its background. In particular, the target may have high or low frequencies compared with the background and/or a different mean or variance than the background. This assumption differs from that of many available target-detection algorithms which rely on only a change of intensity and/or variance of the target from the background.

Our preliminary results in this approach to target detection are discussed in the previous Semiannual Technical Summary,[1] where we addressed this problem in one dimension to develop the theory and our intuition before extending the results to the two-dimensional problem.

In Sec. 2.1, we shall review and elaborate on this one-dimensional theory. In particular, we derive nonrecursive and recursive least-squares procedures for parameter tracking and detection when the data are seen through a finite-extent sliding window. From these derivations, we see that the recursive approach is preferable (computationally) and also gain further insight into the results of Ref. 1. Furthermore, these results lead us to some extensions of Ref. 1 for the one-dimensional problem which appear to hold promise for the two-dimensional case. Specifically, we introduce the use of prediction error in detecting anomalous areas of a signal. A target is detected when a large error occurs in predicting a sample of the signal from its past (i.e., causal prediction). Almost always, the prediction error compares favorably with the change in filter parameters when used in detecting small anomalous areas of a signal. We also investigate the prediction of a signal sample from its surrounding values (i.e., noncausal prediction). Such an approach (noncausal) to detection may be particularly useful in a 2-D context where no directionality is assumed.

In Sec. 2.2, we extend a subset of our 1-D results to develop a 2-D target-detection algorithm which relies on tracking the parameters of a continuously space-varying 2-D filter. As in the 1-D case, we explore both recursive and nonrecursive structures for least-squares parameter estimation. Unlike the 1-D case, however, we shall see that it may be desirable (computationally) to perform 2-D least-squares directly (i.e., nonrecursively) at each pixel rather than recursively. Nevertheless, our derivation of a 2-D recursion lends insight into the nature of filter parameter changes across an image.

As before, detection of an anomalous area of an image is based on changes in the estimated filter parameters. We shall show that, because of a lack of image directionality, the choice of the mask of the 2-D filter model can play an important role in accurate estimation of the parameters (and thus their change used in detection). We shall illustrate such variations on a baseline algorithm by detecting small targets (e.g., 4 x 4, 3 x 3, and 2 x 2 pixels) within RADC image data. Finally, we comment on the extension to 2-D of the use of causal and noncausal prediction error in detection. In particular, we conjecture that the 2-D noncausal prediction error may provide a significant improvement over the present algorithm (based on parameter changes) in detecting small objects in a cluttered background such as a forested area. Consequently, investigation of various 2-D prediction-error functions will be a large component of our future research in target detection.

2.1 Target Detection in One Dimension

Here, we present an algorithm for target detection in 1-D. Our analysis is used as a stepping stone to the development of an analogous 2-D algorithm in the following section.

2.1.1 Nonrecursive Parameter Estimation

Suppose that a 1-D sequence s(n) follows an autoregressive model, and thus is generated by a difference equation of the form

$$s(n) = \sum_{k=1}^{P} a(k)\ s(n-k) + w(n) \tag{1}$$

where w(n) is zero-mean white noise. Our objective is to estimate from s(n) the model parameters a(k) for $k \in [1,P]$. Further, suppose that we have available a segment of s(n) for n in the interval $[-P + n_0, n_1]$, that is, $n_1 - n_0 + P + 1$ data points. We then define the error e(n) over the interval $[n_0,n_1]$ (which we assume is N points in duration, i.e., $N = n_1 - n_0 + 1$) by

$$e(n) = s(n) - \sum_{k=1}^{P} a(k)\ s(n-k) \qquad n \in [n_0,n_1] \quad . \tag{2}$$

e(n) can be interpreted as the error in predicting s(n) from its past P samples. Our goal then becomes to minimize the sum of the squared prediction errors given by

$$E[n_1] = \sum_{n=n_0}^{n_1} e^2(n) \quad . \tag{3}$$

To accomplish this minimization, we first place the problem in a matrix algebra framework. In particular, we define the vectors σ and α by

$$\sigma = \begin{array}{c} \left[\begin{array}{c} s(n_0) \\ s(n_0 + 1) \\ \cdot \\ \cdot \\ \cdot \\ s(n_1) \end{array}\right] \end{array} \updownarrow N \qquad \alpha = \begin{array}{c} \left[\begin{array}{c} a(1) \\ a(2) \\ \cdot \\ \cdot \\ \cdot \\ a(P) \end{array}\right] \end{array} \updownarrow P$$

and the matrix S by

$$S = \begin{array}{cc} \multicolumn{1}{c}{\longleftarrow P \longrightarrow} & \\ \left[\begin{array}{ccccc} s(n_0 - 1) & s(n_0 - 2) & . & . \; . & s(n_0 - P) \\ s(n_0) & s(n_0 - 1) & . & . \; . & s(n_0 - P - 1) \\ & \cdot & & & \\ & \cdot & & & \\ & \cdot & & & \\ s(n_1 - 1) & s(n_1 - 2) & . & . \; . & s(n_1 - P) \end{array}\right] & \updownarrow N \end{array} \quad .$$

Thus, the error $E[n_1]$ in Eq. (3) becomes

$$E[n_1] = [S\alpha - \sigma]^T [S\alpha - \sigma] \tag{4}$$

where T denotes matrix transpose. Differentiating Eq. (4) with respect to α to find the least-squares error, we obtain[2]

$$\alpha = [S^T S]^{-1} S^T \sigma \tag{5}$$

where $S^T S$ is a P x P matrix (P being the order of our model). The solution Eq. (5) is often referred to as the <u>covariance</u> <u>method</u> of linear prediction. Thus, the inverse of a P x P matrix is required to compute α. This will require on the order of P^3 operations.[2]

Suppose now that we wish to obtain an estimate of α for many N-point segments and, in particular, over the intervals $[n_0 + m, n_1 + m]$ where $m = 0, 1, 2, \ldots$. In the next section, we derive a method of updating the coefficients α without recomputing the matrix $(S^T S)^{-1}$.

2.1.2 Recursive Parameter Estimation*

Let us suppose that a new data point is given [i.e., $s(n_1 + 1)$] and that one is eliminated [i.e., $s(n_0)$] so that our new data set runs from $n_0 - P + 1$ to $n_1 + 1$. The sum of the squared errors over the interval $[n_0 + 1, n_1 + 1]$ then becomes

$$E[n_1 + 1] = E[n_1] + e^2(n_1 + 1) - e^2(n_0) \tag{6}$$

where $e(n_1 + 1)$ is the error in predicting the new sample $s(n_1 + 1)$:

$$e(n_1 + 1) = s(n_1 + 1) - \sum_{k=1}^{P} a(k)\, s(n_1 + 1 - k) \tag{7a}$$

*The derivation in this section parallels that of Ref. 3, where the data set increases with time.

and $e(n_0)$ is the error in predicting the old sample $s(n_0)$:

$$e(n_0) = s(n_0) - \sum_{k=1}^{p} a(k)\, s(n_0 - k) \quad . \tag{7b}$$

In matrix form, we can write Eq. (6) as

$$E[n_1 + 1] = E[n_1] + [d\alpha - s(n_1 + 1)]^2 - [c\alpha - s(n_0)]^2 \tag{8}$$

where

$$d^T = \left.\begin{bmatrix} s(n_1) \\ s(n_1 - 1) \\ \cdot \\ \cdot \\ \cdot \\ s(n_1 + 1 - p) \end{bmatrix}\right\updownarrow p \qquad c^T = \left.\begin{bmatrix} s(n_0 - 1) \\ s(n_0 - 2) \\ \cdot \\ \cdot \\ \cdot \\ s(n_0 - p) \end{bmatrix}\right\updownarrow p \quad .$$

Differentiating Eq. (8) with respect to α, we obtain[2]

$$\alpha = [S^T S - c^T c + d^T d]^{-1} \, [S^T \sigma + d^T s(n_1 + 1) - c^T s(n_0)] \quad . \tag{9}$$

Now, we can simplify Eq. (9) as

$$\alpha = [R + G^T F]^{-1} \, [S^T \sigma + G^T Q] \tag{10}$$

where

$$G = \overset{\longleftarrow p \longrightarrow}{\left.\begin{bmatrix} d \\ \hline c \end{bmatrix}\right\updownarrow 2} \qquad F = \overset{\longleftarrow p \longrightarrow}{\left.\begin{bmatrix} d \\ \hline -c \end{bmatrix}\right\updownarrow 2}$$

and

$$Q = \begin{bmatrix} s(n_1 + 1) \\ \\ -s(n_0) \end{bmatrix}$$

and where $R = S^T S$. We wish to avoid direct computation of the matrix inverse in Eq. (10). To accomplish this, we rely on the matrix identity given by

$$[A + BC]^{-1} = A^{-1} - A^{-1}B\ [CA^{-1}B + I]^{-1}\ CA^{-1} \qquad (11)$$

where I is the identity matrix. Letting $A = R$, $B = G^T$, and $C = F$ in Eq. (10), we obtain

$$\alpha = [R^{-1} - R^{-1}G^T(FR^{-1}G^T + I)^{-1}\ FR^{-1}]\ [S^T\sigma + G^TQ] \quad . \qquad (12)$$

Note that R^{-1} is assumed known from the previous computation of α. At this point, for clarity, let us change notation so that α becomes $a[n_1 + 1]$, i.e., the filter parameters estimated within the interval $[n_0 + 1, n_1 + 1]$. Then, Eq. (12) can be written as

$$\begin{aligned} a[n_1 + 1] = {} & R^{-1}S^T\sigma - R^{-1}G^T(FR^{-1}G^T + I)^{-1}\ FR^{-1}S^T\sigma \\ & + R^{-1}G^TQ - R^{-1}G^T(FR^{-1}G^T + I)^{-1}\ FR^{-1}G^TQ \quad . \qquad (13) \end{aligned}$$

Noting that $a[n_1] = R^{-1}S^T\sigma$ and defining a "gain matrix" K as

$$K = R^{-1}G^T(FR^{-1}G^T + I)^{-1} \qquad (14)$$

we have

$$\begin{aligned} a[n_1+1] &= a[n_1] - KFa[n_1] + R^{-1}G^{T}Q - KFR^{-1}G^{T}Q \\ &= a[n_1] - KFa[n_1] \\ &\qquad + R^{-1}G^{T}(FR^{-1}G^{T}+I)^{-1}\,(FR^{-1}G^{T}+I)Q - KFR^{-1}G^{T}Q \\ &= a[n_1] - KFa[n_1] + K(FR^{-1}G^{T}+I)Q - KFR^{-1}G^{T}Q \\ &= a[n_1] - KFa[n_1] + KQ \quad . \end{aligned} \tag{15}$$

Finally, we have

$$a[n_1+1] = a[n_1] + K(Q - Fa[n_1]) \tag{16}$$

which provides a recursive means of updating a[n].

Now let us consider the gain matrix K. The matrix computation involved in finding K can be represented as

$$K = P\!\updownarrow\!\overset{\leftarrow P\rightarrow}{\left[R^{-1}\right]}\overset{\leftarrow 2\rightarrow}{\left[G^{T}\right]}\!\updownarrow\! P\ \ \overset{\longleftarrow 2\longrightarrow}{\left[FR^{-1}G^{T}+I\right]^{-1}}\updownarrow 2 \quad . \tag{17}$$

Thus, we need to compute the inverse of a matrix (i.e., $FR^{-1}G^{T}+I$) which is of size 2 x 2, in contrast to the direct approach which requires the inverse of a P x P matrix [entailing $O(P^3)$ operations]. Note from Eq. (12) that R^{-1} can also be recursively updated as

$$R^{-1} \leftarrow R^{-1} - KFR^{-1} \tag{18}$$

which requires some additional matrix multiplications of order P^2. Consequently, the bulk of the computation lies not in the matrix inversion, but in matrix multiplications of order P^2.

2.1.3 Detection Functionals

It was pointed out in the introduction to Sec. 2 that one procedure for detecting anomalous areas of a signal is to detect abrupt changes in the parameter estimates of the time-varying filter. If the model parameters of a stationary background are known, then one obvious choice of a detection functional is the mean squared difference between the known background parameters and the parameter estimates. If the data deviate from the background (i.e., a target is present), this mean squared error should suddenly change to account for the large error that would be encountered in predicting the first few points of the new data set if $a[n_1 + 1]$ didn't change much from $a[n_1]$. Examples of such abrupt changes in this functional were illustrated in Ref. 1.

This section describes and demonstrates with examples alternative detection functionals which do not require background information or assumptions of stationarity.

The Coefficient Change Functional:- To free ourselves from *a priori* information about the background and the stationarity assumption, we first considered a detection functional of the form:

$$C[n_1] = (a[n_1 + 1] - a[n_1])^T (a[n_1 + 1] - a[n_1]) \qquad (19)$$

which is the mean squared difference between two consecutive parameter estimates. We might expect $C[n_1]$ to jump only when our sliding window first hits the target and when the window just leaves the target. Something similar to $C[n_1]$ was considered by Dove and Oppenheim[4] who referred to Eq. (19) as the coefficient change functional.

Although this functional has been used with good success in both 1-D and in extensions to 2-D, a potential problem exists. This change functional is sometimes not responsive to small extent targets in a highly cluttered background (e.g., trees). Note that the parameter estimates rely on the correlation function $R = S^T S$. A small extent target may not change R sufficiently to generate a corresponding change in filter parameters. However, such absence of change in the parameters may actually aid a detection functional based on prediction error - as we show in the following section.

<u>The Prediction Error Functional:-</u> Let us return to Eq. (16) which we now write as

$$a[n_1 + 1] = a[n_1] + ke[n_1] \tag{20}$$

where

$$\begin{aligned} e[n_1] &= Q - Fa[n_1] \\ &= \begin{bmatrix} s(n_1 + 1) \\ \\ -s(n_0) \end{bmatrix} - \begin{bmatrix} d \\ \hline -c \end{bmatrix} a[n_1] \\ &= \begin{bmatrix} s(n_1 + 1) - da[n_1] \\ \\ -s(n_0) + ca[n_1] \end{bmatrix} \\ &= \begin{bmatrix} e_1[n_1] \\ \\ e_2[n_1] \end{bmatrix} . \end{aligned} \tag{21}$$

From Eqs. (7) and (8), we see that $e_1(n)$ in Eq. (21) is the error in predicting the new signal sample with the old parameters. When this new signal sample $s(n_1 + 1)$ is a member of an anomalous area, we expect the error in predicting $s(n_1 + 1)$ to change abruptly when the old set of coefficients $a[n_1]$ is based on data not encompassing the target. Note that, although the coefficient change $a[n_1 + 1] - a[n_1]$ is a function of $e_1(n)$ (i.e., $a[n_1 + 1] - a[n_1] = Ke[n_1]$), the time-varying gain term K and $e_2(n)$ may drown out the change in $e_1(n)$.

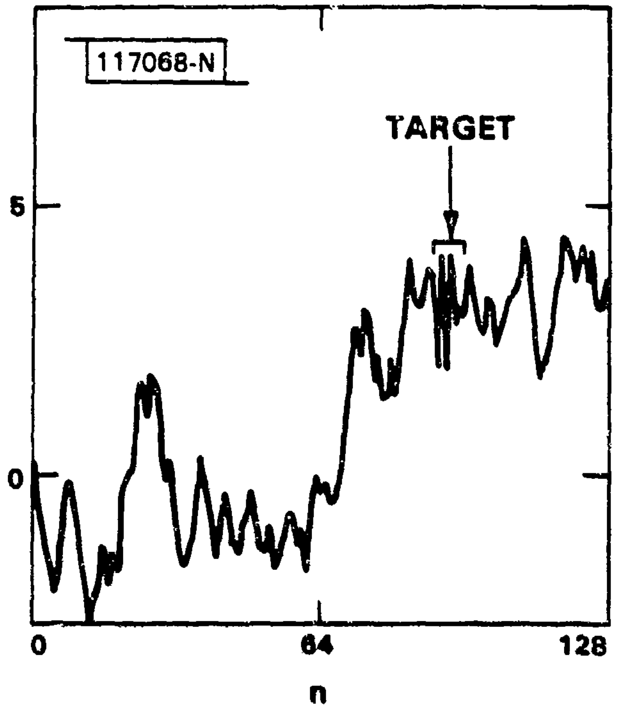

Fig. 1. A target and signal based on a first-order model.

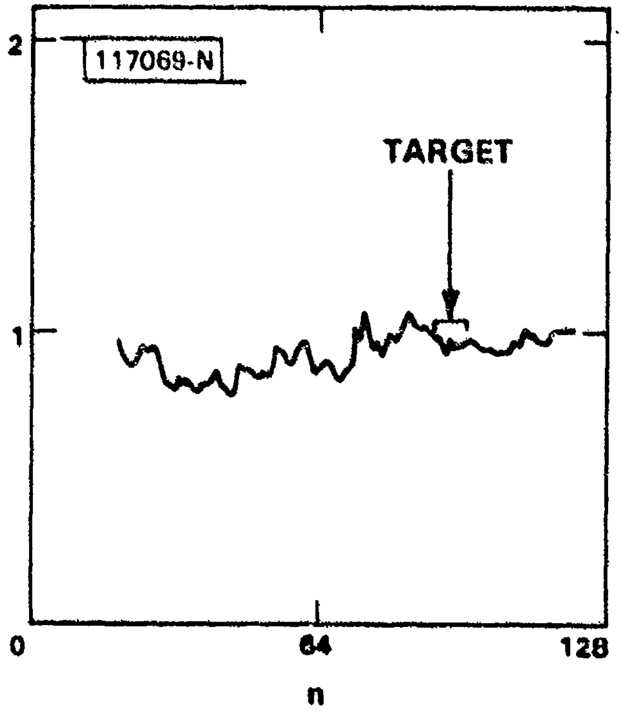

Fig. 2(a). Coefficient estimate for Fig. 1.

It is possible to generalize this notion of prediction error for target detection. In particular, $e_1(n)$ in Eq. (21) represents a causal prediction error, i.e., $s(n_1 + 1)$ is predicted from the past P values of s(n). Clearly, it is possible to formulate the least-squares estimation problems of Secs. 2.1.1 and 2.1.2 in terms of noncausal prediction of a sample of s(n) from its surrounding neighbors, i.e., a smoothing of s(n) from points to its left and right:

$$s(n) = \sum_{\substack{k=-Q \\ k\neq 0}}^{P} a(k)\ s(n-k) + w(n) \quad . \tag{22}$$

We have applied such noncausal detection to 1-D with success. Noncausal prediction may be particularly useful in 2-D detection where there exists no apparent directionality. In the next section, we compare by example the various detection functionals discussed in this section.

2.1.4 Examples

Consider a sequence s(n) of the form:

$$s(n) = 0.95s(n-1) + w(n) \tag{23}$$

where w(n) is zero-mean white noise. A sample function of s(n) is shown in Fig. 1. A 4-point high-frequency target at n = 90, 91, 92, and 93 is also shown there. Figure 2(a) depicts the coefficient estimate based on a 16-point window, and Fig. 2(b) shows the corresponding coefficient change C[n]. Note that a global change in mean and other local phenomena in the background has been emphasized over the target. Figure 2(c) depicts the squared prediction error, $e_1^2[n]$; the target is clearly detected.

In a second example, we analyzed a cross section of pure tree data from the field-tree RADC image. Figure 3 depicts this cross section with its mean removed (see Sec.2.1.5), with a 3-point constant-level target imposed at n = 64, 65, and 66. A 5-pole model was assumed and a 16-point sliding window was used. C[n] and $e_1^2[n]$ are shown in Figs. 4(a) and (b), respectively. Note that $e_1^2[n]$ is without the false alarm of C[n].

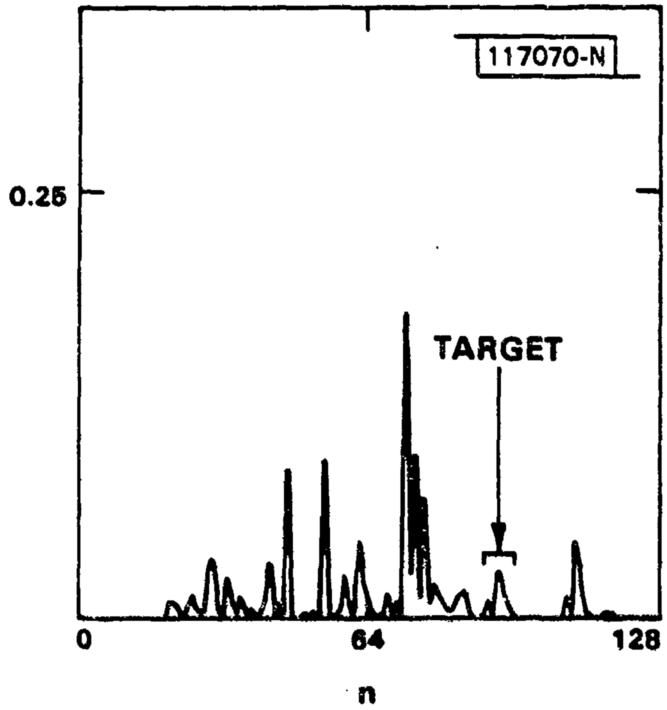

Fig. 2(b). C[n] for Fig. 1.

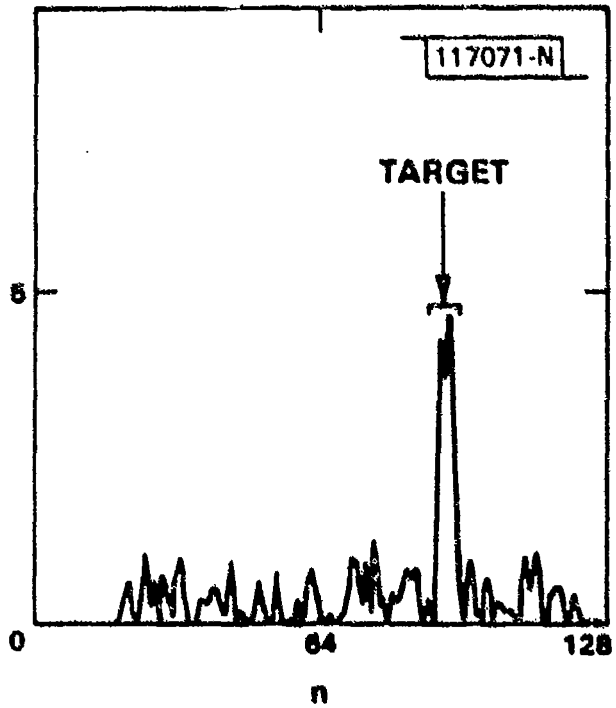

Fig. 2(c). $e_1^2[n]$ for Fig. 1.

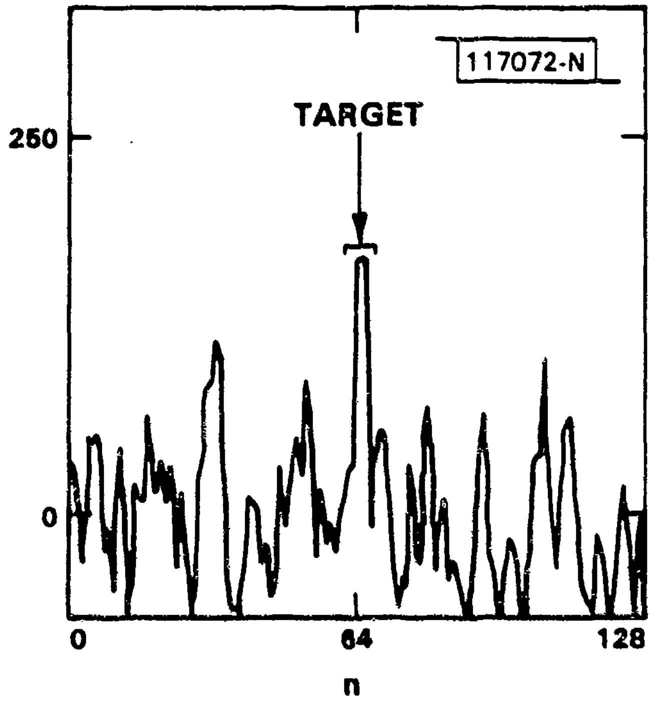

Fig. 3. Cross section of RADC data.

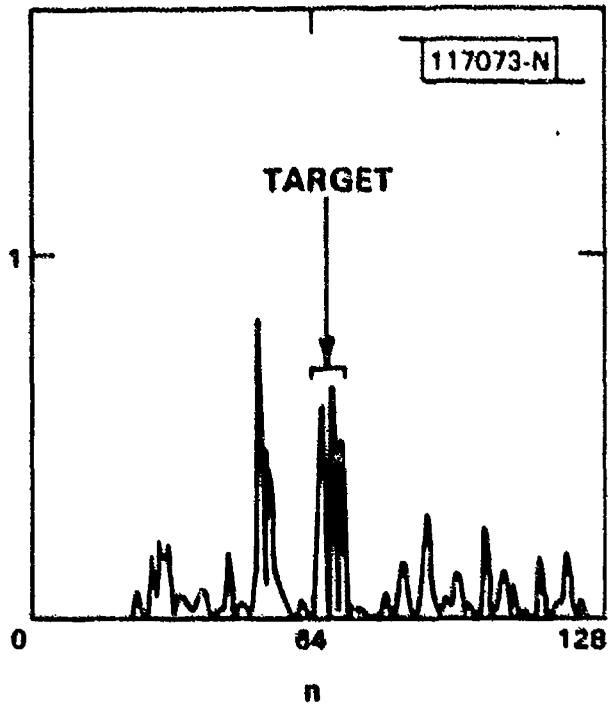

Fig. 4(a). C[n] for Fig. 3.

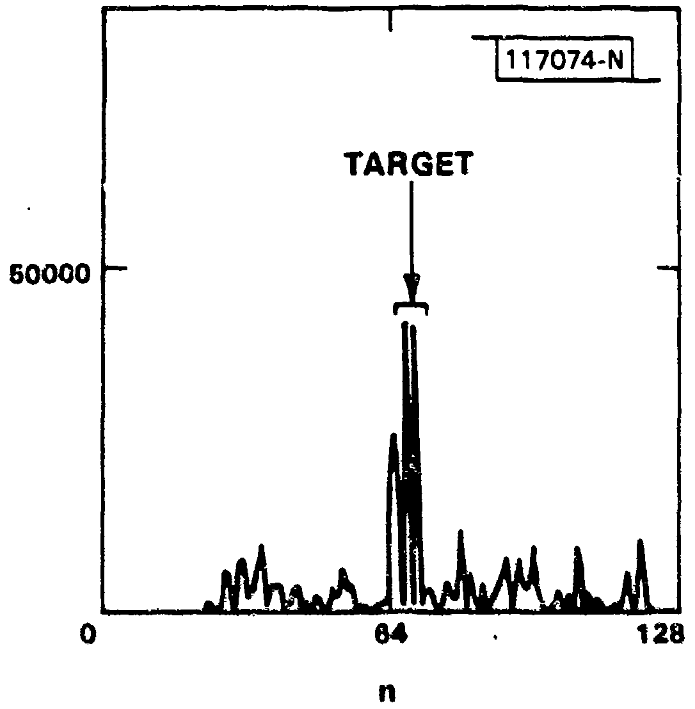

Fig. 4(b). $e_1^2[n]$ for Fig. 3.

Consider again the sequence generated by Eq. (23) in Fig. 1, but now where we assume a model of the form

$$s(n) = a(1)\ s(n-1) + a(2)\ s(n+1) + w(n) \tag{24}$$

i.e., s(n) is a function of its two nearest adjacent neighbors. Figure 5 depicts the noncausal prediction error corresponding to the signal in Fig. 1 and which uses a 16-point sliding window. Finally, the noncausal model Eq. (24) was used to generate a prediction error from the image cross section given in Fig. 3. The prediction error is depicted in Fig. 6. The target is clearly detected in both cases.

2.1.5 Removal of the Mean Level

It should be noted that the image cross section of Fig. 3 contains a nonzero mean which violates our model Eqs. (1) and (22) where zero-mean white

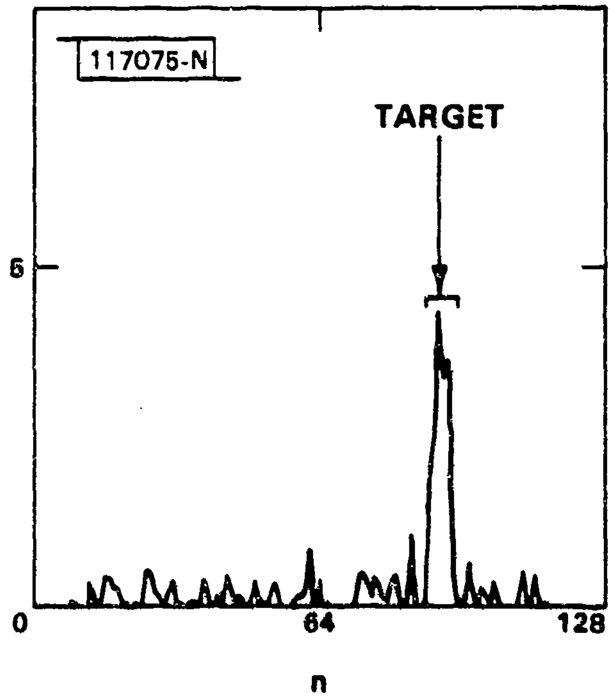

Fig. 5. $e_1^2[n]$ (noncausal) for Fig. 1.

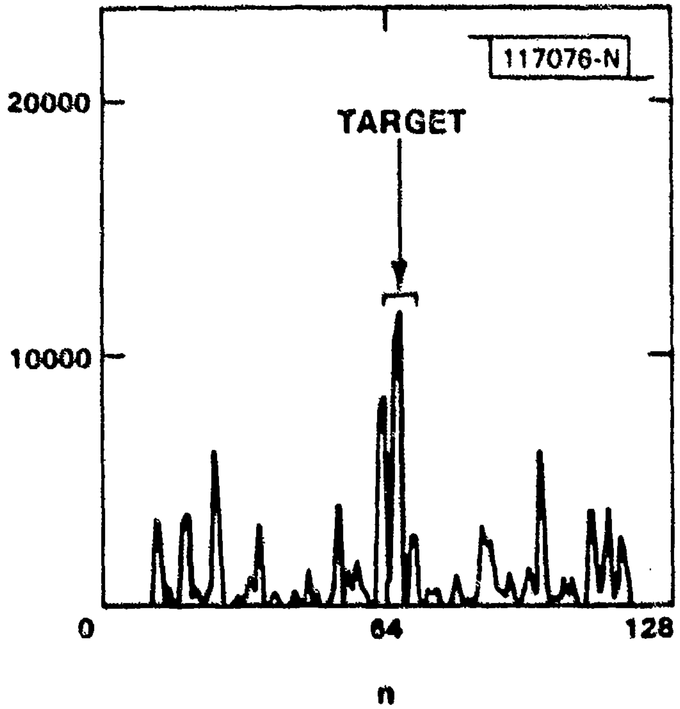

Fig. 6. $e_1^2[n]$ (noncausal) for Fig. 3.

noise is assumed to drive a linear system. One approach to addressing a time-varying nonzero-mean level is to incorporate it within our model so that we estimate the mean as well as the parameters a(k):

$$s(n) = \sum_{k=1}^{p} a(k)\ s(n - k) + w(n) + C \tag{25}$$

where C is a mean level added to the input w(n). The output will then contain a nonzero mean.

Now, if we perform least-squares estimation, we find that the best estimate of the a(k)'s is given approximately by (this derivation parallels that in Sec. 2.1.1):

$$\alpha = (S^T S\sigma - m_s^2)^{-1}\ (S^T\sigma - m_s^2) \tag{26}$$

where N is the data length and

$$m_s = \sum_{n=n_0}^{n_1} s(n)/N \tag{27}$$

which is an estimate of the mean of the output. The best estimate of C is given approximately by

$$C = m_s\left[1 - \sum_{k=1}^{p} a(k)\right] \quad . \tag{28}$$

Equation (26) simply implies that we first estimate the mean of s(n) through Eq. (27), subtract it from s(n), and then solve Eq. (5) of Sec. 2.1.1. To find C, we "inverse filter" the mean at the output to obtain the mean at the input.

Removal of the mean may not only lead to more accurate determination of the a(k)'s, but also provide an additional parameter in our functional for target detection. We shall discuss this possibility in the next section.

2.2 Target Detection in Two Dimensions

One obvious approach to extending our 1-D results to 2-D target detection is to apply 1-D recursive least-squares estimation along rows or columns of an image. Of course, such a procedure is not "optimal" since the correlation of a sequence s(n,m) is considered in only one direction. We have found empirically two problems with this approach: (1) various detection functionals used have a large dynamic range (i.e., the functional exhibits little consistency from line to line), and (2) an undesirably long window length (along each row or column) may be required, resulting in excessive "memory" of the functional.[5,6]

In this section, we describe a least-squares procedure which invokes rows and columns simultaneously, i.e., we address the 2-D problem directly. After considering both nonrecursive and recursive estimation, we devise a 2-D target detection algorithm which is a significant improvement over its line-by-line counterpart. We demonstrate the algorithm by applying it to the detection of anomalous areas within aerial photographs obtained from RADC.

2.2.1 Nonrecursive Parameter Estimation

Conceptually, 2-D least-squares estimation is similar to the 1-D case. We begin with a 2-D sequence which follows an autoregressive model, so it is given by a difference equation of the form

$$s(n,m) = \sum_{\substack{j \\ (j,k)\neq(0,0)}} \sum_{k} a(j,k)\; s(n-j,\, m-k) + w(n,m) \tag{29}$$

where w(n,m) is zero-mean white noise. For the moment, we shall assume that the prediction coefficients a(j,k) fall within a (P x Q) first-quadrant mask, as illustrated in Fig. 7. For simplicity, we limit our derivations to this class of prediction masks, although it is clearly applicable to more general mask shapes. In particular, we shall investigate second-quadrant and nonsymmetric half-plane masks in Sec. 2.2.5.

Our objective is to estimate from s(n,m) the model parameters a(j,k) for $j = 0,1,\ldots,P$ and $k = 0,1,\ldots,Q$, with $j = k \neq 0$. Further, let us suppose

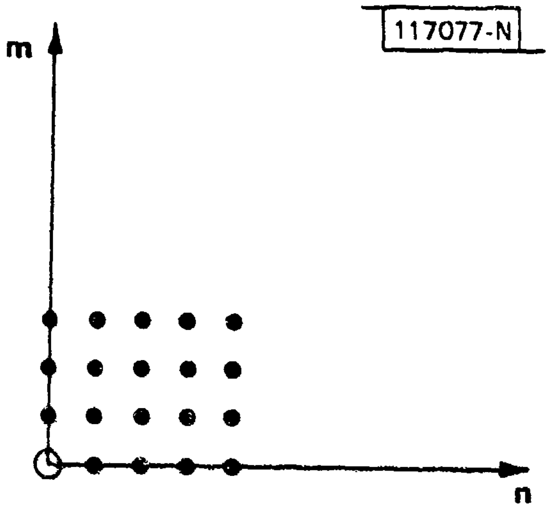

Fig. 7. A first-quadrant quarter-plane mask for predictor coefficients a[n,m].

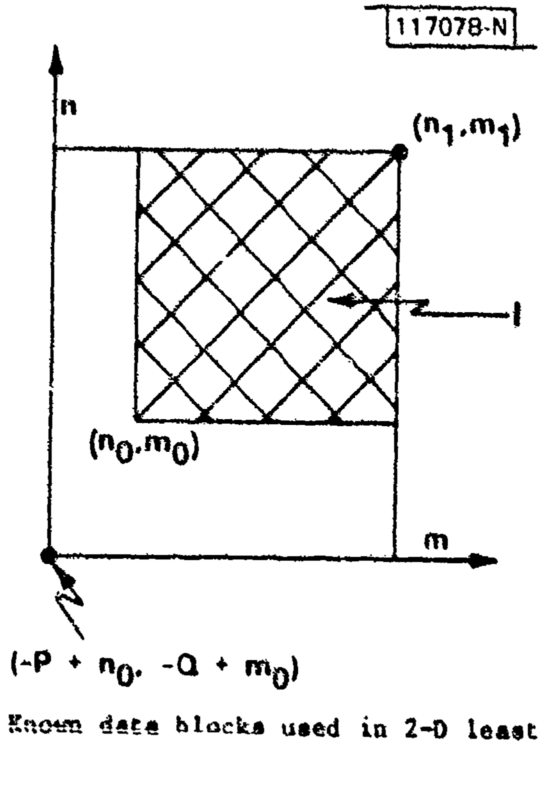

Fig. 8. Known data blocks used in 2-D least squares.

that we have available a square block of data, i.e., $s(n,m)$ for $(n,m) \varepsilon [-P + n_0,n_1] \times [-Q + m_0,m_1]$ (see Fig. 8). We then define the error $e(n,m)$ over the region I, given by $I = [n_0,n_1] \times [m_0,m_1]$, as

$$e(n,m) = s(n,m) - \sum_{\substack{j=0 \\ j=k\neq 0}}^{P} \sum_{k=0}^{Q} a(j,k)\ s(n - j, m - k) \qquad (n,m) \varepsilon I \quad . \tag{30}$$

Our goal becomes to minimize the sum of the squared errors given by

$$E[n_1,m_1] = \sum_{n=n_0}^{n_1} \sum_{m=m_0}^{m_1} e^2(n,m) \quad . \tag{31}$$

The approach we take is to transform the 2-D problem to a 1-D problem so that a 1-D least-squares solution is applicable. Note, however, that we will still have solved the 2-D least-squares problem. In particular, we wish to transform Eq. (31) into a form similar to the 1-D error expression in Eq. (6). To accomplish this transformation, we define the following vectors $s[n_1,m_1]$ and σ by

$$a[n_1,m_1] = \left[\begin{array}{c} a(0,1) \\ a(0,2) \\ \vdots \\ a(0,Q) \\ \hline a(1,0) \\ a(1,1) \\ \vdots \\ a(1,Q) \\ \hline \vdots \\ \hline a(P,0) \\ a(P,1) \\ \vdots \\ a(P,Q) \end{array}\right] \Big\updownarrow (PQ - 1) \qquad \sigma = \left[\begin{array}{c} s(n_0,m_0) \\ s(n_0,\ m_0 + 1) \\ \vdots \\ s(n_0,m_1) \\ \hline s(n_0 + 1,\ m_0) \\ s(n_0 + 1,\ m_0 + 1) \\ \vdots \\ s(n_0 + 1,\ m_1) \\ \hline \vdots \\ \hline s(n_1,m_0) \\ s(n_1,\ m_0 + 1) \\ \vdots \\ s(n_1,m_1) \end{array}\right] \Big\updownarrow (PQ - 1) \tag{32}$$

and the matrix S by

$$S = \begin{array}{cc} \longleftarrow PQ - 1 \longrightarrow & \\ \left[\begin{array}{c} \text{———} A_0 \text{———} \\ \text{———} A_1 \text{———} \\ \vdots \\ \text{———} A_{N-1} \text{———} \end{array}\right] & \Big\updownarrow N^2 \end{array} \tag{33a}$$

where

$$A_j = \overbrace{\begin{bmatrix} [s(n_0 + j - 0, m_0 - 1)\ldots s(n_0 + j - 0, m_0 + Q)] & \ldots & [s(n_0 + j - P, m_0 - 0)\ldots s(n_0 + j - P, m_0 - Q)] \\ [s(n_0 + j - 0, m_0 + 1 - 1)\ldots s(n_0 + j - 0, m_0 + 1 + Q)] & \ldots & [s(n_0 + j - P, m_0 + 1 - 0)\ldots s(n_0 + j - P, m_0 + 1 - Q)] \\ & \vdots & \\ [s(n_0 + j - 0, m_1 - 1)\ldots s(n_0 + j - 0, m_1 - Q)] & \ldots & [s(n_0 + j - P, m_1 - 0)\ldots s(n_0 + j - P, m_1 - Q)] \end{bmatrix}}^{PQ - 1} \Big\updownarrow N \tag{33b}$$

and where we have assumed the known data segment to be of extent N x N. Note that σ is a vector consisting of the concatenation of the rows of $s(n,m)$ over I, $a[n_1,m_1]$ is a vector consisting of the concatenation of the rows of $a(j,k)$ for $(j,k) \in [0,P] \times [0,Q]$ with $j = k \neq 0$, and S is a matrix which consists of the concatenation of rows of various sub-sequences of the known $s(n,m)$ required in predicting each value of $s(n,m)$ over I.

Therefore, we can write Eq. (31) as

$$E[n_1,m_1] = \sum_{n=n_0}^{n_1} \sum_{m=m_0}^{m_1} e^2(n,m)$$

$$= (Sa[n_1,m_1] - \sigma)^T (Sa[n_1,m_1] - \sigma) \quad . \tag{34}$$

From the similarity of Eq. (34) with Eq. (4), we can write the solution to minimizing Eq. (34) with respect to $a[n_1,m_1]$ as

$$a[n_1,m_1] = R^{-1} S^T \sigma \tag{35a}$$

where

$$R = S^T S \quad . \tag{35b}$$

Note that the matrix R is of extent $(PQ - 1) \times (PQ - 1)$ and difficulty in its inversion is dependent on the model order, not on the size of the known block of data.

In particular, since R is generally not Toepltz, its inversion will require on the order of $(PQ)^3$ operations (i.e., multiplications within, for example, Cholesky's decomposition algorithm[2]). It should be pointed out that the computation of R^{-1} can probably be reduced by considering its block-like structure resulting from the conversion of a 2-D problem to a 1-D problem. Thus, assuming $P,Q << N$, the bulk of the computation is embedded within forming $R = S^T S$ which requires on the order of N^4 operations.

2.2.2 Recursive Parameter Estimation

Let us suppose now that we wish to estimate the parameters $a[n_1,m_1]$ over a large data set. Our goal in this section is to devise a recursive procedure for updating the parameter estimates over the 2-D plane without recomputing the inverse of R. Unlike the 1-D problem, many ways exist to recurse in the 2-D plane, i.e., vertically, horizontally, diagonally, etc. In all these procedures, entire rows and/or columns of data enter and exit the view of a 2-D sliding window. An alternative to this class of recursions is the class of "snake recursions" which allow only one or a few signal samples to enter and exit the window's field of view. Such recursions may not only reduce computation, but provide greater resolution in detecting small-extent 2-D targets. For the present, however, we have investigated the row (column) recursion, and its derivation is the primary emphasis of this section. In the conclusion of this section, however, we briefly describe one point-by-point recursion, its computational advantages, and its potential for high-resolution detection.

Now let us suppose that we add a new row of data, i.e., $s(n_1 + 1, m)$ for $m \in [-Q + m_0, m_1]$ and eliminate the known row at $n = -P + n_0$ (see Fig. 9). Our goal is to generate the coefficient estimates $a[n_1 + 1, m_1]$ based on the new row of data and the old coefficient estimates $a[n_1,m_1]$. The new error of interest is given by:

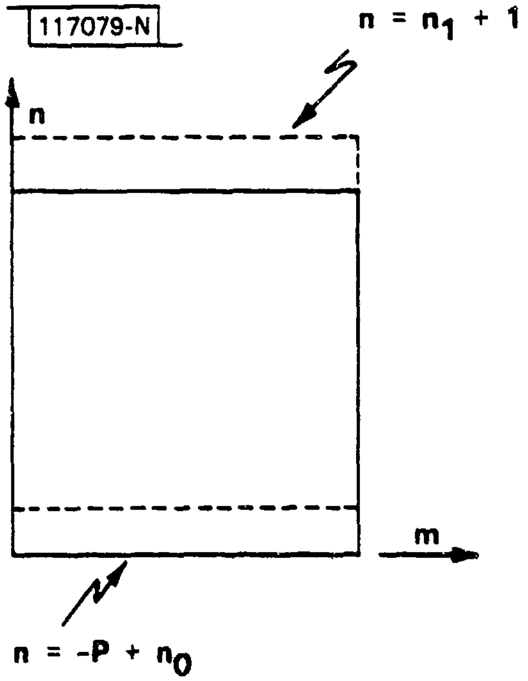

Fig. 9. Data update used in 2-D recursive least squares.

$$E[n_1 + 1, m_1] = E[n_1, m_1] - \sum_{m=m_0}^{m_1} e^2(n_0, m) + \sum_{m=m_0}^{m_1} e^2(n_1 + 1, m) \tag{36}$$

where $e(n_1 + 1, m)$ is the error in predicting the new row $s(n_1 + 1, m)$:

$$e(n_1 + 1, m) = s(n_1 + 1, m) - \sum_{\substack{j=0 \\ (j,k)\neq(0,0)}}^{P} \sum_{k=0}^{Q} a(j,k)\, s(n_1 + 1 - j, m - k) \tag{37}$$

and where $e(n_0, m)$ is the error in predicting the old row:

$$e(n_0, m) = s(n_0, m) - \sum_{\substack{j=0 \\ (j,k)\neq(0,0)}}^{P} \sum_{k=0}^{Q} a(j,k)\, s(n_0 - j, m - k) \quad . \tag{38}$$

In matrix form, we can write Eq. (36) as

$$E[n_1 + 1, m_1] = E[n_1,m_1] + (Da[n_1 + 1, m_1] - g)^T (Da[n_1 + 1, m_1] - g)$$

$$- (Ca[n_1 + 1, m_1] - f)^T (Ca[n_1 + 1, m_1] - f) \qquad (39)$$

where $a[n_1 + 1, m_1]$ is given in Eq. (32), and where g and f are the vector representations of the old and new rows, respectively:

$$g = \left[\begin{array}{c} s(n_0,m_0) \\ s(n_0, m_0 + 1) \\ \cdot \\ \cdot \\ \cdot \\ s(n_0,m_1) \end{array}\right] \Big\updownarrow N \qquad f = \left[\begin{array}{c} s(n_1 + 1, m_0) \\ s(n_1 + 1, m_0 + 1) \\ \cdot \\ \cdot \\ \cdot \\ s(n_1 + 1, m_1) \end{array}\right] \Big\updownarrow N \qquad (40)$$

and where $C = A_0$ and $D = A_N$ with A_j the N x (PQ - 1) matrix given in Eq. (33b). Noting the similarity of Eq. (40) with Eq. (8), we can write the solution $a[n_1 + 1, m_1]$ which minimizes $E[n_1 + 1, m_1]$ as

$$a[n_1 + 1, m_1] = [R - C^TC + D^TD]^{-1} [S^T\sigma + D^Tg - C^Tf] \quad . \qquad (41)$$

Now, we can simplify Eq. (41) as

$$a[n_1 + 1, m_1] = [R - G^TF]^{-1} [S^T\sigma + C^TQ] \qquad (42)$$

where

$$G = \begin{array}{c} \xleftrightarrow{PQ - 1} \\ \left[\begin{array}{c} D \\ \hline C \end{array}\right] \end{array} \Big\updownarrow 2N \qquad F = \begin{array}{c} \xleftrightarrow{PQ - 1} \\ \left[\begin{array}{c} D \\ \hline -C \end{array}\right] \end{array} \Big\updownarrow 2N$$

and where

$$Q = \begin{array}{cc} \leftarrow 1 \rightarrow & \\ \left[\begin{array}{c} g \\ \hline -f \end{array}\right] & \updownarrow 2N \end{array} .$$

Following our derivation in Sec. 2.1.2, we then have the recursive computation of $a[n_1 + 1, m_1]$:

$$a[n_1 + 1, m_1] = a[n_1,m_1] + K(Q - Fa[n_1,m_1]) \tag{43}$$

where K is given by Eq. (14) and R^{-1} can be recursively updated as in Eq. (18).

The matrix computation involved in finding K can be represented as (with $L = PQ - 1$):

$$K = \updownarrow L \overset{\leftarrow L \rightarrow}{\left[R^{-1}\right]} \overset{\leftarrow 2N \rightarrow}{\left[G^T\right]} \updownarrow L \quad \overset{\longleftarrow 2N \longrightarrow}{\left[FR^{-1}G^T + I\right]^{-1}} \updownarrow 2N \tag{44}$$

so that K is an L x 2N matrix. The important point here is that the inverse of a 2N x 2N matrix is required to compute K, implying a number of operations on the order of $(N)^3$ operations per image pixel - clearly intolerable for windows of large extent. Thus, unlike in the 1-D case, a recursion based on Eq. (43) may entail more computation than performing 2-D least-squared directly through Eq. (35). Of course, we have not considered any special structure in $FR^{-1}G^T + I$ which may help to reduce computation.

An alternative to this row-by-row recursion (which we promised to briefly describe before leaving this section) is a "snake-like" point-by-point recursion illustrated in Fig. 10. To compute the parameter estimates under the window in Fig. 9, we must perform N "little" recursions each of which is 1-D in nature, thus requiring the inverse of N 2 x 2 matrices [see Eq. (14)]. Therefore, we are confronted with the trade-off between one

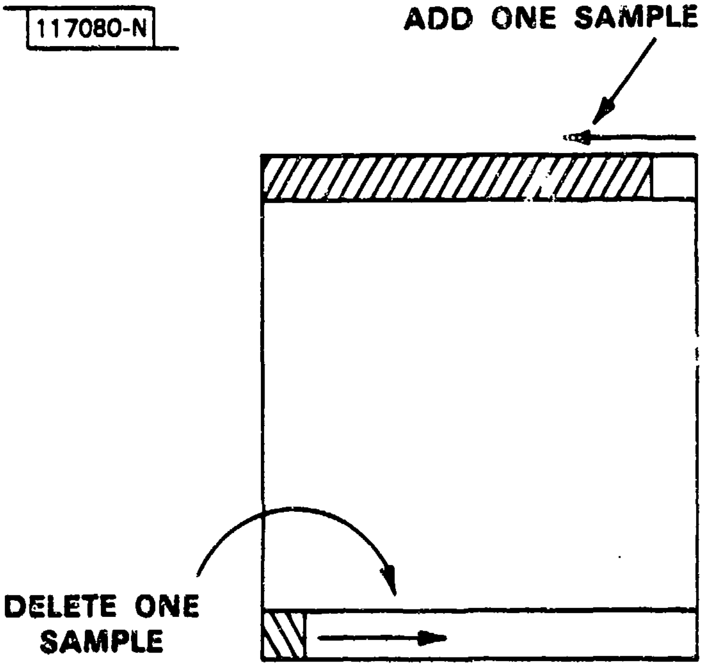

Fig. 10. A point-by-point 2-D recursion.

inverse of a large 2N x 2N matrix and N inverses of little 2 x 2 matrices - clearly a significant reduction in computation. One other potential advantage of this point-by-point recursion is that it does not attempt to predict an entire row on each step. An implication of this property is that if target detection relies on a prediction error, a point-by-point recursion may be more likely to detect very small-extent and/or closely spaced targets. However, since N recursions are required for each pixel, our detection function becomes three-dimensional. Such speculation serves as a starting point for future research.

For the present, however, we have put aside all recursive solutions and devised a detection algorithm based on direct computation of Eq. (35). Nevertheless, all this discussion about recursions is useful since (as we shall see in the following section) it has provided insight into the nature of changes in the parameter estimates across an image.

2.2.3 Detection Functionals

Clearly, we can extend our 1-D detection functionals to 2-D. The functional we have chosen to use (for the present) in the implementation of 2-D detection algorithms is the coefficient change functional given by

$$C[n_1,m_1] = (a[n_1 + 1, m_1] - a[n_1,m_1])^T (a[n_1 + 1, m_1] - a[n_1,m_1]) \ . \qquad (45)$$

From Eq. (43), $C[n_1,m_1]$ can be written as

$$C[n_1,m_1] = \{K(Q - Fa[n_1,m_1])\}^T \{K(Q - Fa[n_1,m_1])\}$$

$$= K\left[\begin{array}{c} e_1[n_1,m_1] \\ \hline e_2[n_1,m_1] \end{array}\right]^T \times K\left[\begin{array}{c} e_1[n_1,m_1] \\ \hline e_2[n_1,m_1] \end{array}\right] \qquad (46)$$

where $e_1[n_1,m_1]$ is the error in predicting the new row with the old coefficient estimates:

$$e_1[n_1,m_1] = Da[n_1,m_1] - g \qquad (47a)$$

and $e_2[n_1,m_1]$ is the error in predicting the old row with the old coefficient estimates:

$$e_2[n_1,m_1] = Ca[n_1,m_1] - f \quad . \qquad (47b)$$

Thus, $C[n_1,m_1]$ resembles a weighted sum of these prediction errors. It is important to note that since Eqs. (35) and (43) must yield the same result [based, of course, on the same particular known segment of $s(n,m)$], in either case (i.e., for either the recursive or nonrecursive solution), $C[n_1,m_1]$ can be written as Eq. (46).

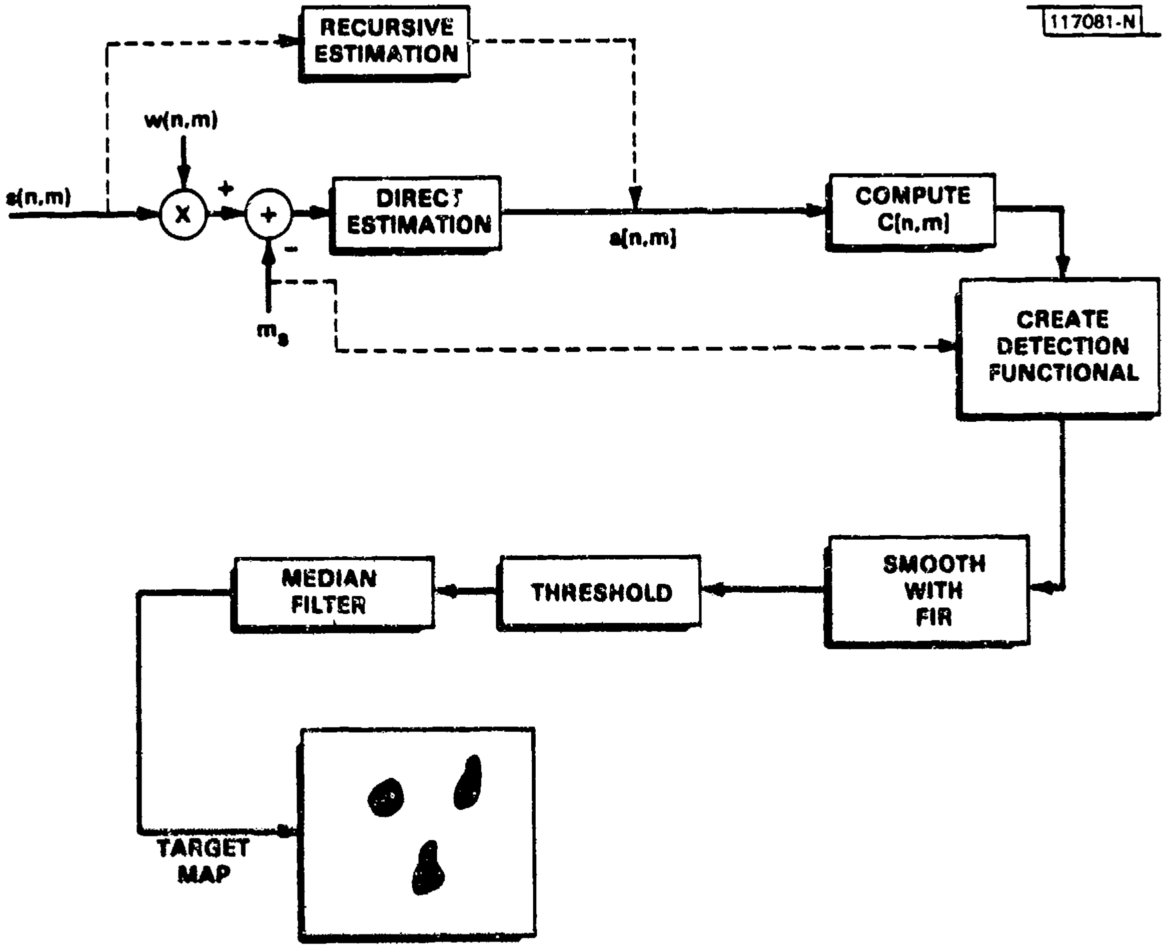

Fig. 11. 2-D detection algorithm.

An alternative detection functional [analogous to the 1-D function $e_1[n_1]$ in Eq. (21)] is the prediction error $e_1[n_1,m_1]$. At each pixel of an image, we might consider abrupt changes in $e_1[n_1,m_1]$. If a target happens to fall along the new row which just enters the 2-D (sliding) window's view, we expect a large error in prediction at the target location along that row. For similar arguments given in Sec. 2.1.3 for the 1-D case, this prediction error may be more sensitive to targets of small extent than the coefficient change function. Furthermore, in the case of images, noncausal prediction may be more reasonable to attempt than the causal prediction given above. Judging from our 1-D experiments of Sec. 2.1.4, detection based on $e_1[n_1,m_1]$ (or its noncausal counterpart) holds promise, and thus serves as a possible avenue of future research. However, in the remainder of this report, we shall investigate the use of the coefficient change function for detection.

2.2.4 The 2-D Target-Detection Algorithm

Our overall 2-D target-detection algorithm based on the coefficient change functional is illustrated in Fig. 11. Hidden within this block diagram are a number of important decisions. First, m_s represents the local mean of s(n,m) under the sliding window w(n,m). As in the 1-D case (see Sec. 2.1.5), in theory this mean level should be removed, although in practice it was found to make little difference in the coefficient change functional. (However, the coefficient estimates themselves may differ.) Thus, m_s was not subtracted from the data, although it was found useful at times as an additional parameter within the detection functional.

Another decision (hidden within the "direct estimation" box) involves the choice of the shape and size of the prediction mask. Recall that our derivations in the previous sections were restricted to a first-quadrant quarter-plane mask. Since an image has no apparent directionality, a quarter-plane mask may bias the coefficient estimates. Furthermore, from the work of Ekstrom and Woods,[7] we know that a quarter-plane mask (even infinite in extent) is not sufficient to match an arbitrary power spectrum, whereas a nonsymmetric half-plane mask is sufficient.

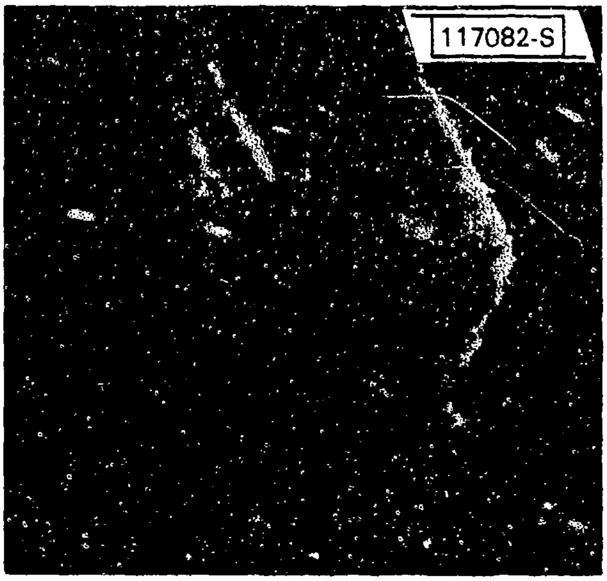

Fig. 12. RADC image.

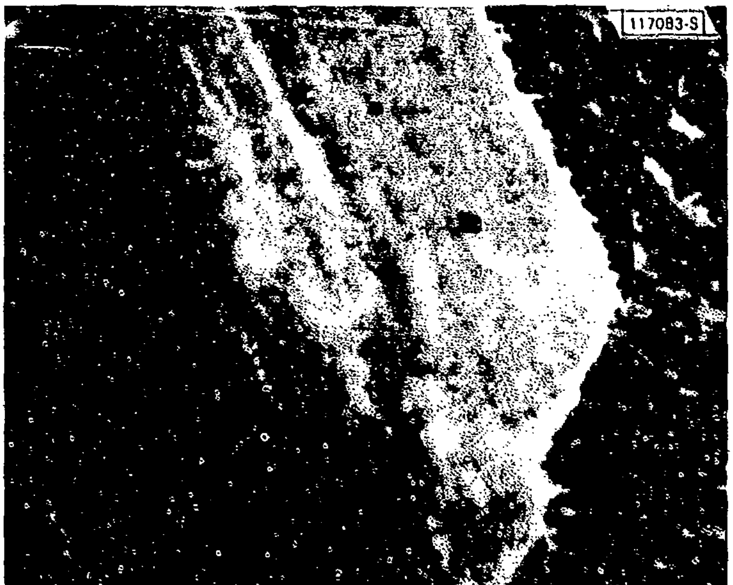

Fig. 13. RADC image with targets.

As illustrated in Fig. 11, before applying a threshold, we smoothed the detection functional with an FIR filter. This operation helped reduce the number of false alarms. The threshold choice was based on heuristics, although we are currently considering an adaptive threshold (e.g., CFAR used in radar). Finally, to create the target map, a 3 x 3 median filter was applied to further reduce the false-alarm rate.

The following examples illustrate these considerations and also demonstrate the success of detecting targets of small extent.

2.2.5 Examples

In the following set of examples, we have applied the 2-D nonrecursive least-squares algorithm to the RADC image data which are composed of forests and fields illustrated in Fig. 12. Anomalous regions in the photograph were created by taking small areas (2 x 2, 3 x 3, and 4 x 4 pixels in extent) of forested regions and transplanting them to the field regions as depicted in Fig. 13. The coefficient-change technique can find these anomalous areas in spite of the use of the very low-order model with $P = Q = 1$ (i.e., three independent coefficients).

Figure 14 illustrates the coefficient change function Eq. (45) (before smoothing) based on a 5 x 5 sliding window* and a 2 x 2 first-quadrant prediction mask. Note that $C[n,m]$ is largest at the three targets and at the field-field and field-tree transitions. Figure 15 depicts $c[n,m]$ based on a 2 x 2 second-quadrant prediction mask. The average of Figs. 14 and 15 is shown in Fig. 16 which appears to be an improvement over either of these two functionals, i.e., the target and transition regions are more clearly accentuated. A cross section of the original image (through the 2 x 2 target) and the corresponding cross section of the averaged $C[n,m]$ are illustrated in Figs. 17(a) and (b), respectively.

We might consider the average in Fig. 16 to be a rough approximation to a functional based on a nonsymmetric half-plane (NSHP) prediction mask.[7] A comparison of a first-quadrant, second-quadrant, and a 12-coefficient NSHP prediction mask is illustrated in Fig. 18. $C[n,m]$ associated with this NSHP

*All examples in this section are based on a 5 x 5 sliding window.

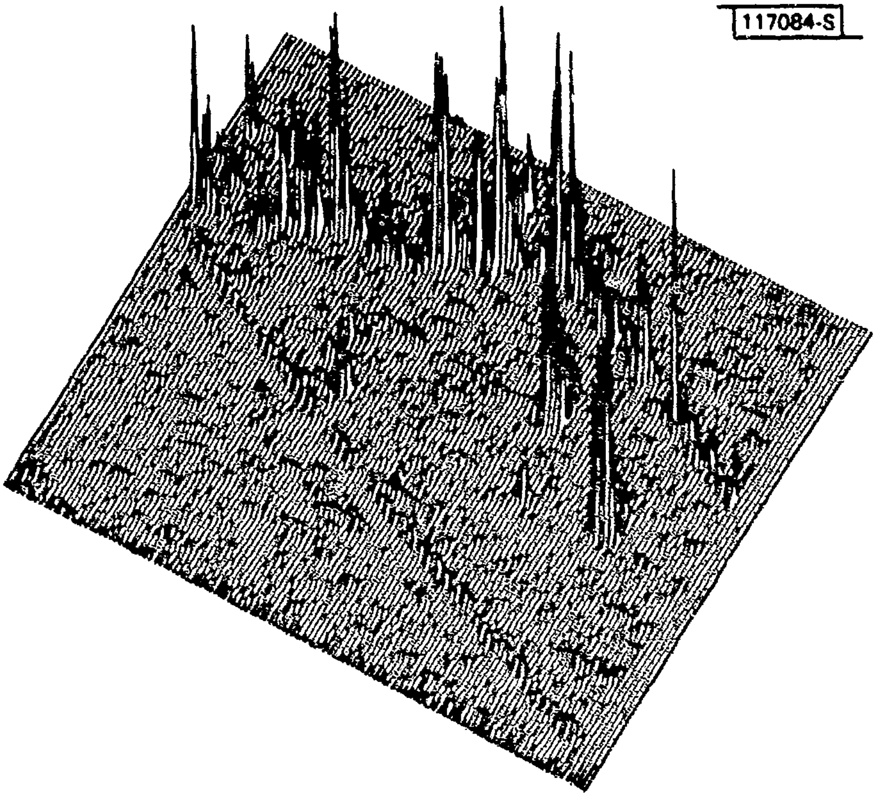

Fig. 14. Detection functional based on first-quadrant mask.

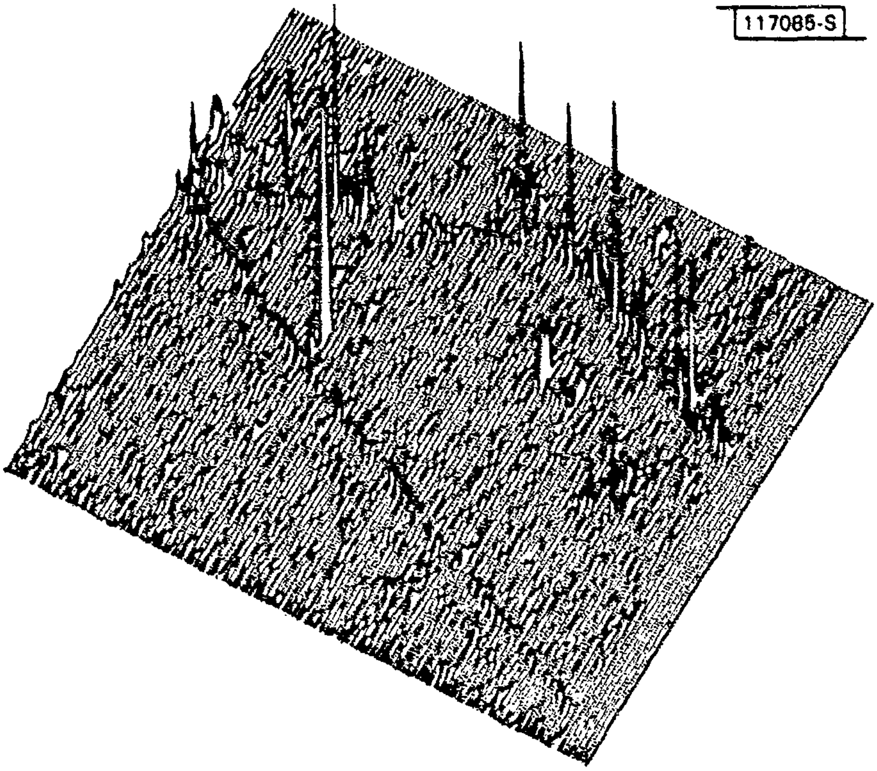

Fig. 15. Detection functional based on second-quadrant mask.

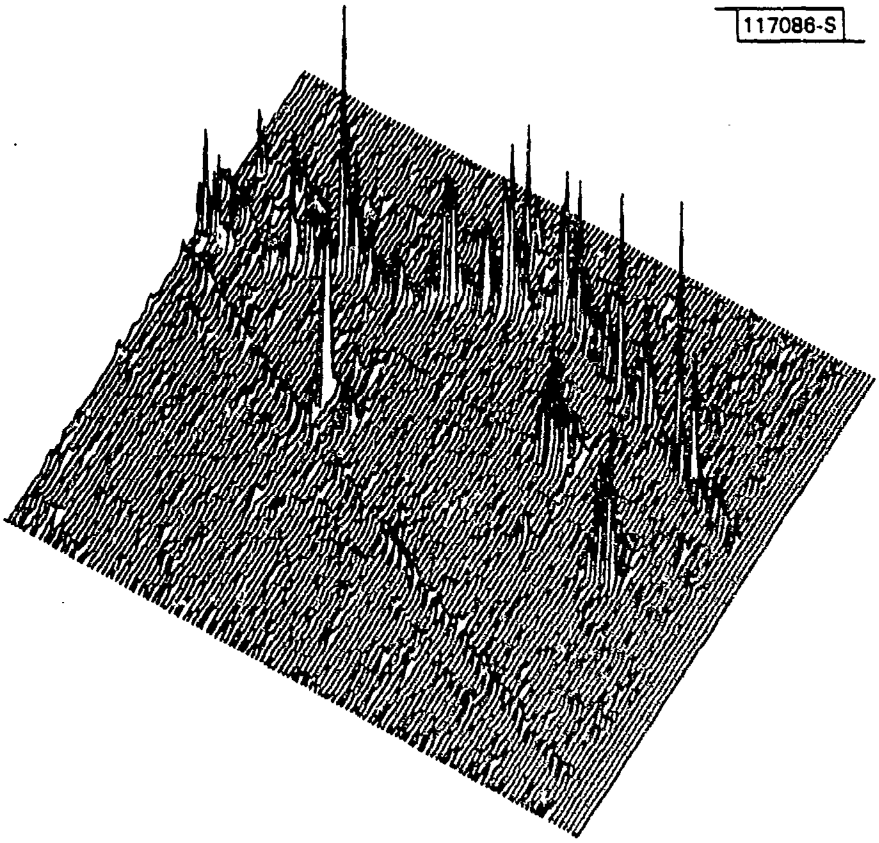

Fig. 16. Average of Figs. 14 and 15.

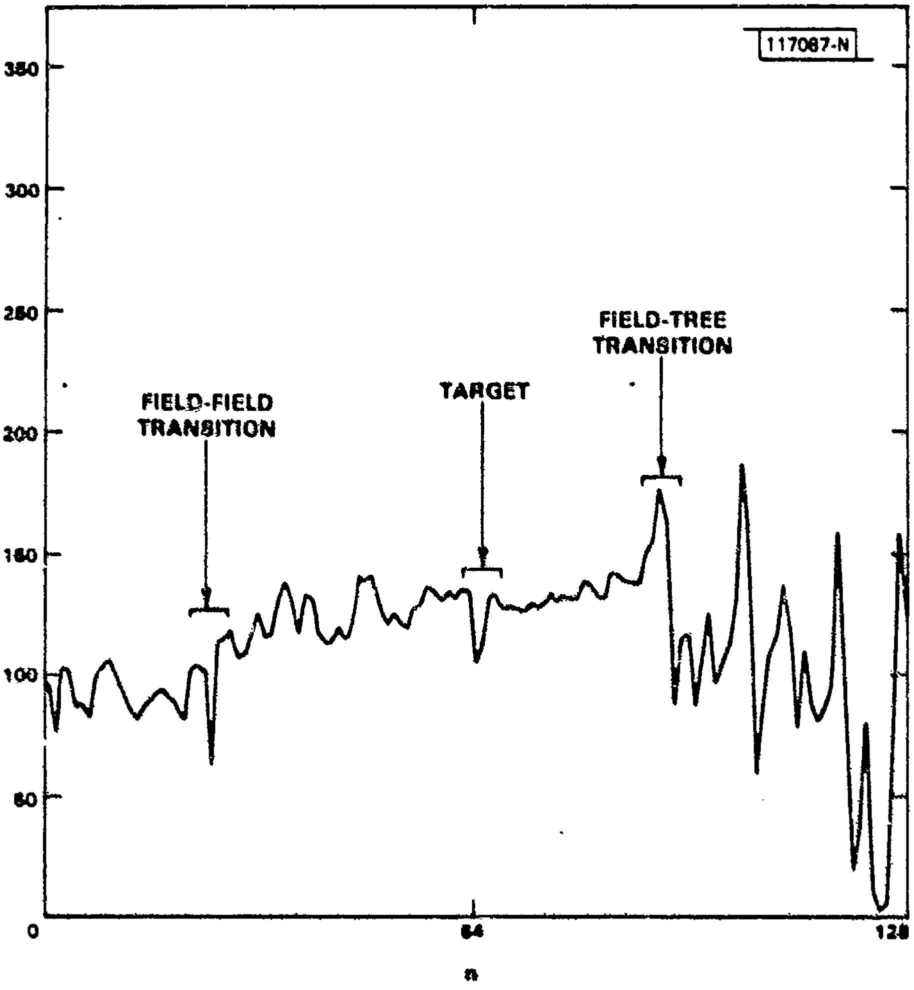

Fig. 17(a). Cross section of RADC image in Fig. 13.

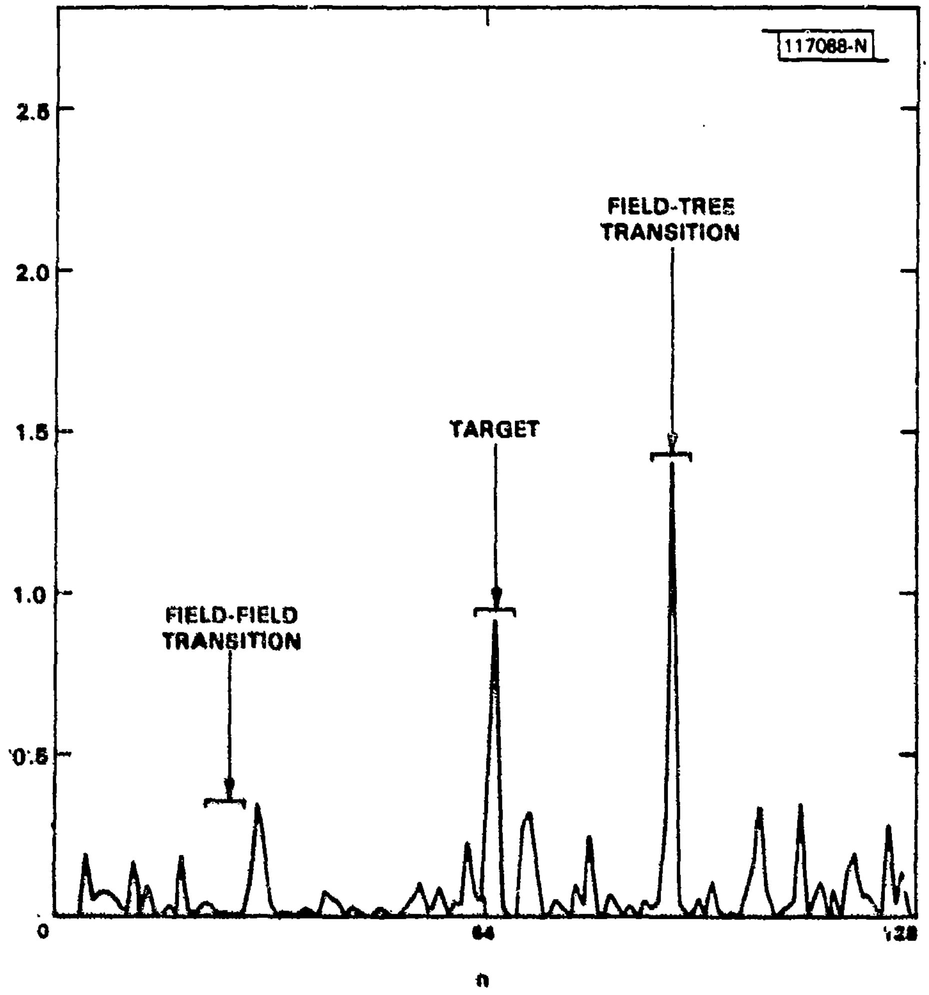

Fig. 17(b). Cross section of detection functional in Fig. 16.

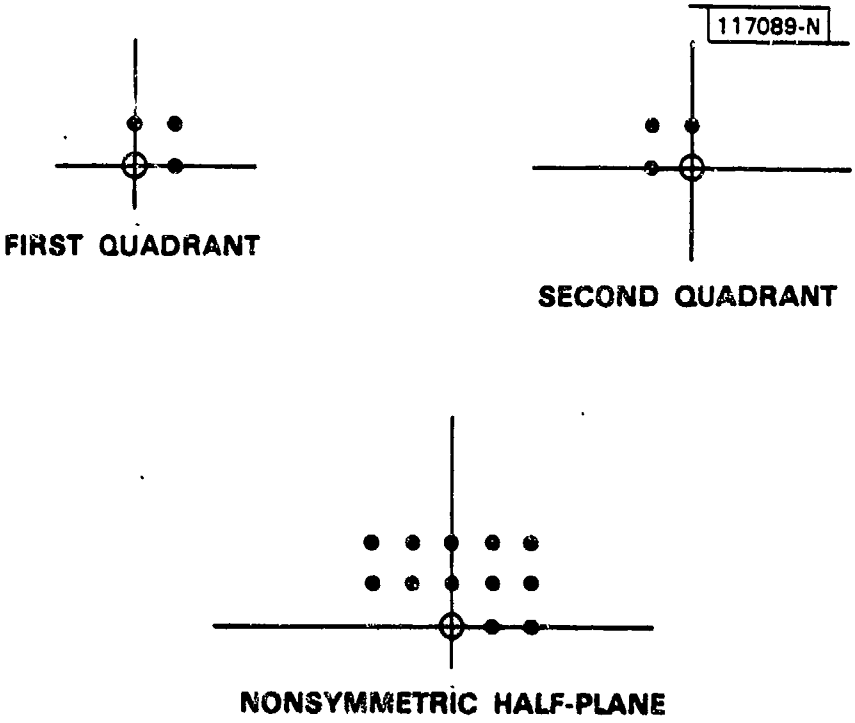

Fig. 18. Comparison of different prediction masks.

mask performed about as well as the average depicted in Fig. 16 in detecting targets, although it seems to exhibit a poorer performance in detecting the field-field transition.

Figures 19(a) and (b) depict the smoothed C[n,m] of Fig. 16 after application of a 3 x 3 FIR smoothing filter. The result of thresholding and median filtering to create the target map is shown in Fig. 20 superimposed on the original image. All three targets and the field-tree transition are clearly detected.

These examples represent a subset of a large number of experiments performed in 2-D based on the coefficient-change function. For example, the local mean level was not used in the experiments described above, but was found in other examples to enhance C[n,m] which was reflected in reducing the false-alarm rate of the target map. Other experiments involved smoothing the parameters $a[n_1,m_1]$ with a Laplacian before computing $C[n_1,m_1]$ (represents a differencing operation in two directions), applying an exponential weight to the windowed data before estimation (in order to emphasize certain regions),

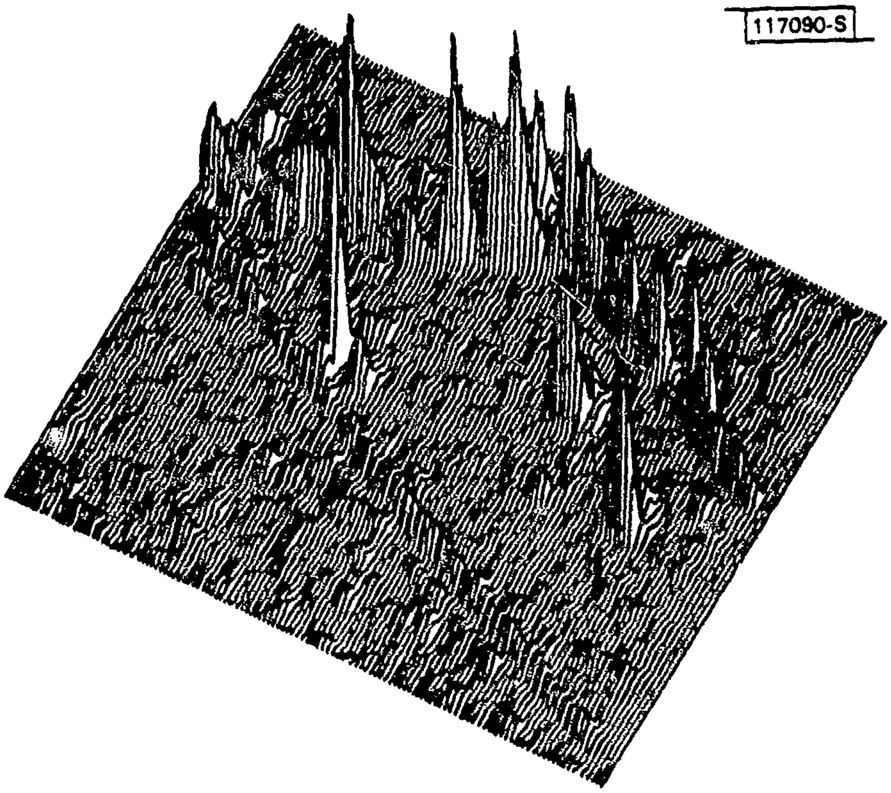

Fig. 19(a). Smoothed detection functional for RADC image.

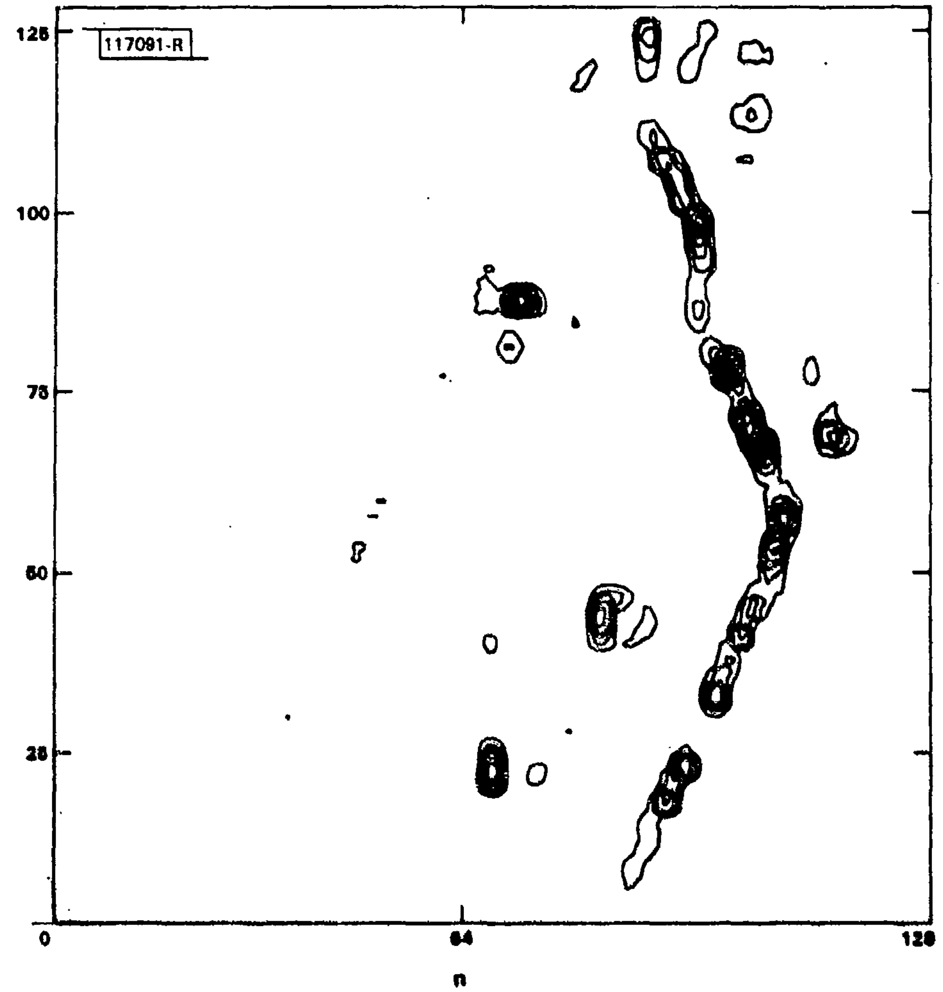

Fig. 19(b). Contour map of detection functional in Fig. 19(a).

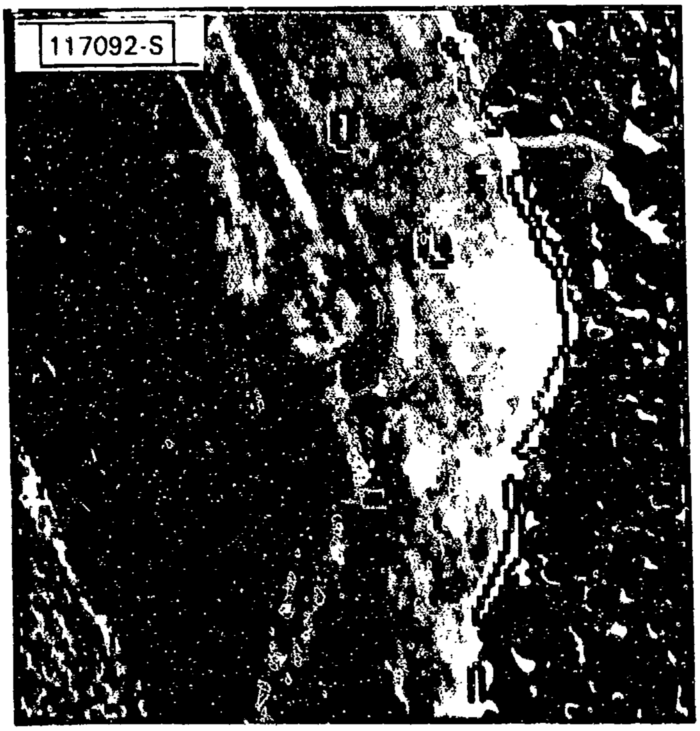

Fig. 20. Target map corresponding to detection functional and image in Figs. 19 and 13.

increasing and decreasing the window size, basing a detection functional on a geometric mean squared change (rather than an arithmetic mean), incorporating a variance measurement into the detection functional, etc. One problem in all these studies involved the detection of anomalous areas in the tree data of Fig. 12. Preliminary results indicate, however, that a 2-D extension of the use of the prediction error may have some advantages in such detection problems over the coefficient-change function.

REFERENCES

1. Semiannual Technical Summary, Multi-Dimensional Signal Processing Research Program, Lincoln Laboratory, M.I.T. (30 September 1981), DTIC AD-A111971.

2. J.H. McClellan, "Parametric Signal Modeling Notes," Course Notes for Advanced Topics in Digital Signal Processing (1981).

3. D.B. Harris, "Recursive Solution of the Covariance Normal Equations," Course Notes for Advanced Topics in Digital Signal Processing (1978).

4. W.P. Dove and A.V. Oppenheim, "Event Location Using Recursive Least Squares Signal Processing," IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing, Denver, Colorado, 9-11 April 1980.

5. T.F. Quatieri, "Target Detection by Recursive Least Squares Filtering," Lincoln Laboratory MDSP Memo (September 1981).

6. T.F. Quatieri, "Target Detection by 2-D Recursive Least Squares Estimation," Lincoln Laboratory MDSP Memo (November 1981).

7. M.P. Ekstrom and J.W. Woods, "Two-Dimensional Spectral Factorization with Applications in Recursive Digital Filtering," IEEE Trans. Acoust., Speech, and Signal Processing ASSP-24, 115-128 (1976).

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER ESD-TR-82-041	2. GOVT ACCESSION NO. AD-A118186	3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle) Multi-Dimensional Signal Processing Research Program	5. TYPE OF REPORT & PERIOD COVERED Semiannual Technical Summary 1 October 1981 – 31 March 1982
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Dan E. Dudgeon	8. CONTRACT OR GRANT NUMBER(s) F19628-80-C-0002
9. PERFORMING ORGANIZATION NAME AND ADDRESS Lincoln Laboratory, M.I.T. P.O. Box 73 Lexington, MA 02173-0073	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Program Element No. 62702F Project No. 4594
11. CONTROLLING OFFICE NAME AND ADDRESS Rome Air Development Center Griffiss AFB, NY 13440	12. REPORT DATE 31 March 1982
	13. NUMBER OF PAGES 48
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Electronic Systems Division Hanscom AFB Bedford, MA 01731	15. SECURITY CLASS. (of this report) Unclassified
	15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

None

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

image classification
iterative image restoration
target detection
image segmentation
iterative filtering

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This Semiannual Technical Summary covers the period 1 October 1981 through 31 March 1982. It describes the significant results of the Lincoln Laboratory Multi-Dimensional Signal Processing Research Program sponsored by the Rome Air Development Center, in the areas of image segmentation, classification, target detection, and adaptive contrast enhancement.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)